

● PRINTER RUSH ●

(PTO ASSISTANCE)

Application : <u>09/640 602</u>	Examiner : <u>Ferris</u>	GAU : <u>U663</u>
From : <u>NIB</u>	Location : <u>IDC</u> FMF FDC	Date : <u>7-28-05</u>
Tracking #: <u>06075454-6</u>		Week Date: <u>2-7-05</u>

DOC CODE	DOC DATE	MISCELLANEOUS
<input type="checkbox"/> 1449	_____	<input type="checkbox"/> Continuing Data
<input type="checkbox"/> IDS	_____	<input type="checkbox"/> Foreign Priority
<input type="checkbox"/> CLM	_____	<input type="checkbox"/> Document Legibility
<input type="checkbox"/> IIFW	_____	<input type="checkbox"/> Fees
<input type="checkbox"/> SRFW	_____	<input type="checkbox"/> Other
<input checked="" type="checkbox"/> DRW	<u>8-17-00</u>	
<input type="checkbox"/> OATH	_____	
<input type="checkbox"/> 312	_____	
<input type="checkbox"/> SPEC	_____	

[RUSH] MESSAGE: Attn: Chief Draftsperson

All 28 Figures have lines through drawings.

Please Resolve.

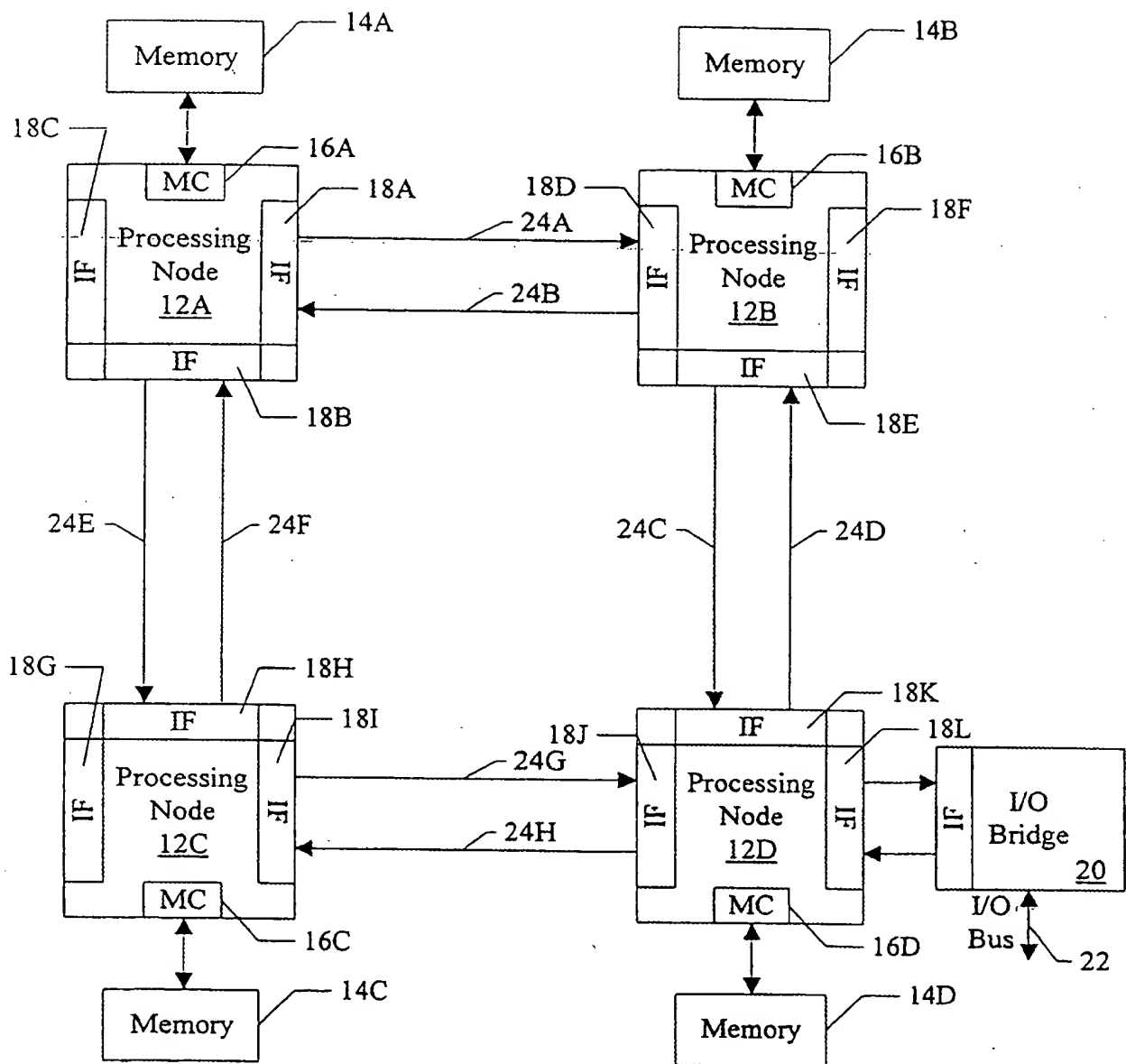
Thank you
8/1/03

[XRUSH] RESPONSE: _____

Drawing corrected

INITIALS: [Signature]

NOTE: This form will be included as part of the official USPTO record, with the Response document coded as XRUSH.
REV 10/04



10

Fig. 1

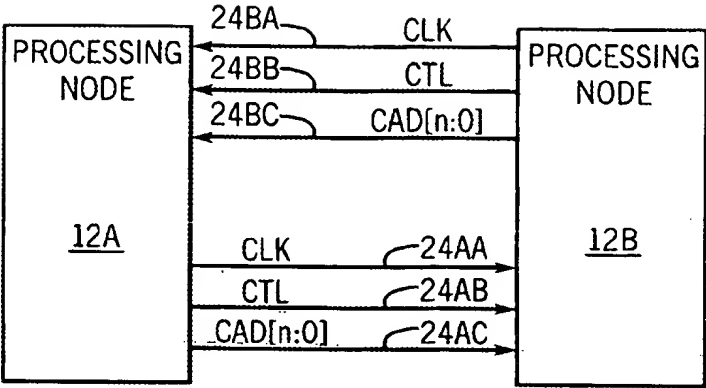


FIG. 2

Bit Time	7	6	5	4	3	2	1	0
0	RSV		CMD[5:0]					
1	RespData [1:0]		Response [1:0]		PostCmd Data[1:0]		PostCmd [1:0]	
2	RSV		Probe [1:0]		NonPost Data[1:0]		NonPost Cmd[1:0]	
3	RSV							

180

FIG. 18

Bit Time	7	6	5	4	3	2	1	0
0			CMD[5:0]					
1								
2								
3								

30

FIG. 3

Bit Time	7	6	5	4	3	2	1	0
0	Src Unit [1:0]		CMD[5:0]					
1	DestNode[2:0]			Dest Unit[1:0]		SrcNode[2:0]		
2				SrcTag[4:0]				
3	Addr[7:2]							
4	Addr[15:8]							
5	Addr[23:16]							
6	Addr[31:24]							
7	Addr[39:32]							

32

FIG. 4

Bit Time	7	6	5	4	3	2	1	0
0	Src Unit [1:0]		CMD[5:0]					
1	DestNode[2:0]			Dest Unit[1:0]		SrcNode[2:0]		
2				SrcTag[4:0]				
3	Sh							

34 ↗

FIG. 5

Bit Time	7	6	5	4	3	2	1	0
0	Data[7:0]							
1	Data[15:8]							
2	Data[23:16]							
3	Data[31:24]							
4	Data[39:32]							
5	Data[47:40]							
6	Data[55:48]							
7	Data[63:56]							

36 ↗

FIG. 6

<u>CMD Code</u>	<u>VChan</u>	<u>Command</u>	<u>Packet Type</u>
000000	—	Nop	Info
000001	NPC	VicBlk	Request / Address / Data
000010	—	Reserved	—
000011	NPC	ValidateBlk	Request / Address
000100	NPC	RdBlk	Request / Address
000101	NPC	RdBkS	Request / Address
000110	NPC	RdBkMod	Request / Address
000111	NPC	ChangeToDirty	Request / Address
x01xxx	NPC or PC	Wr(Sized)	Request / Address / Data
01xxx	NPC	Read(Sized)	Request / Address
100xxx	—	Reserved	—
110000	R	RdResponse	Response / Data
110001	R	ProbeResp	Response
110010	R	TgtStart	Response
110011	R	TgtDone	Response
110100	R	SrcDone	Response
110101	R	MemCancel	Response
11011x	—	Reserved	—
11100x	P	Probe	Request / Address
11101x	P	Broadcast	Request / Address
11110x	—	Reserved	—
111110	—	Reserved	—
111111	—	Sync	Info

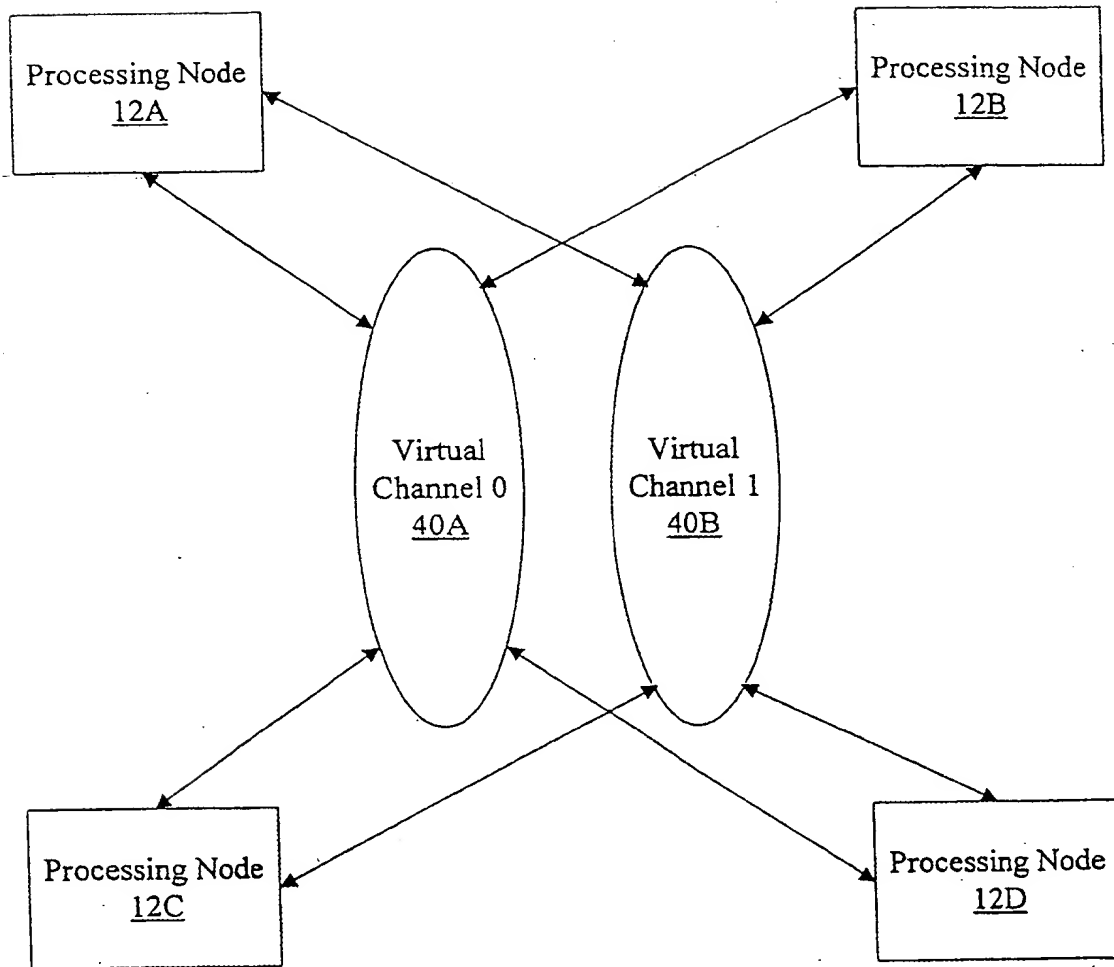


Fig. 8

Virtual Channels

<u>Virtual Channel</u>	<u>Applicable Links</u>
Posted Commands	Coherent and NonCoherent
Non-Posted Commands	Coherent and NonCoherent
Responses	Coherent and NonCoherent
Probes	Coherent Only

42



Fig. 9

Block diagram of a network node 12A. The node includes a Main Controller (MC) 16A at the top. Below it is Packet Processing Logic 58, which is connected to MC 16A via a bidirectional arrow. Packet Processing Logic 58 is also connected to two interfaces: IF 18C on the left and IF 18A on the right, both via bidirectional arrows. Below Packet Processing Logic 58 are two processing paths. The left path consists of a Cache 54 (dashed box) connected to a Processor Core 56 (dashed box) via a bidirectional arrow. The right path consists of a Cache 50 (solid box) connected to a Processor Core 52 (solid box) via a bidirectional arrow. Both Cache 54 and Cache 50 are connected to Packet Processing Logic 58 via bidirectional arrows. At the bottom of the node is Interface 18B, which is connected to both Processor Core 56 and Processor Core 52 via bidirectional arrows. The entire node is labeled 12A at the bottom left.

Fig. 10

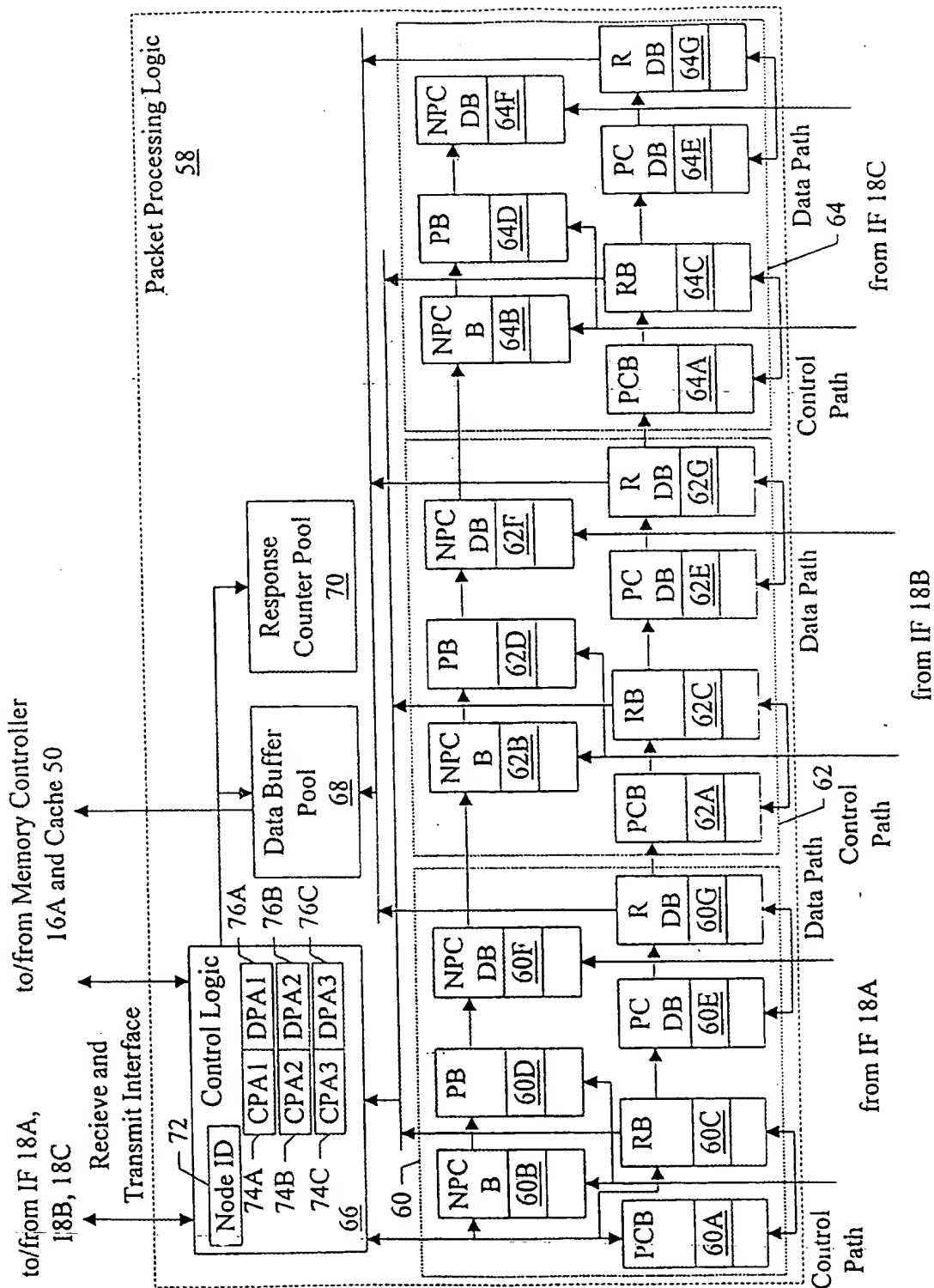


Fig. 11

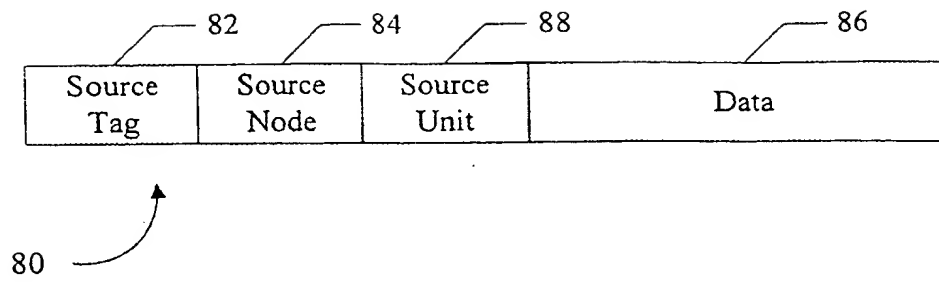


Fig. 12

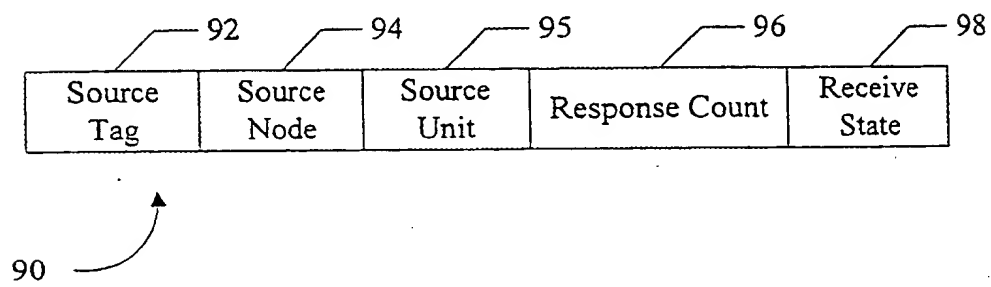


Fig. 13

FIG. 14

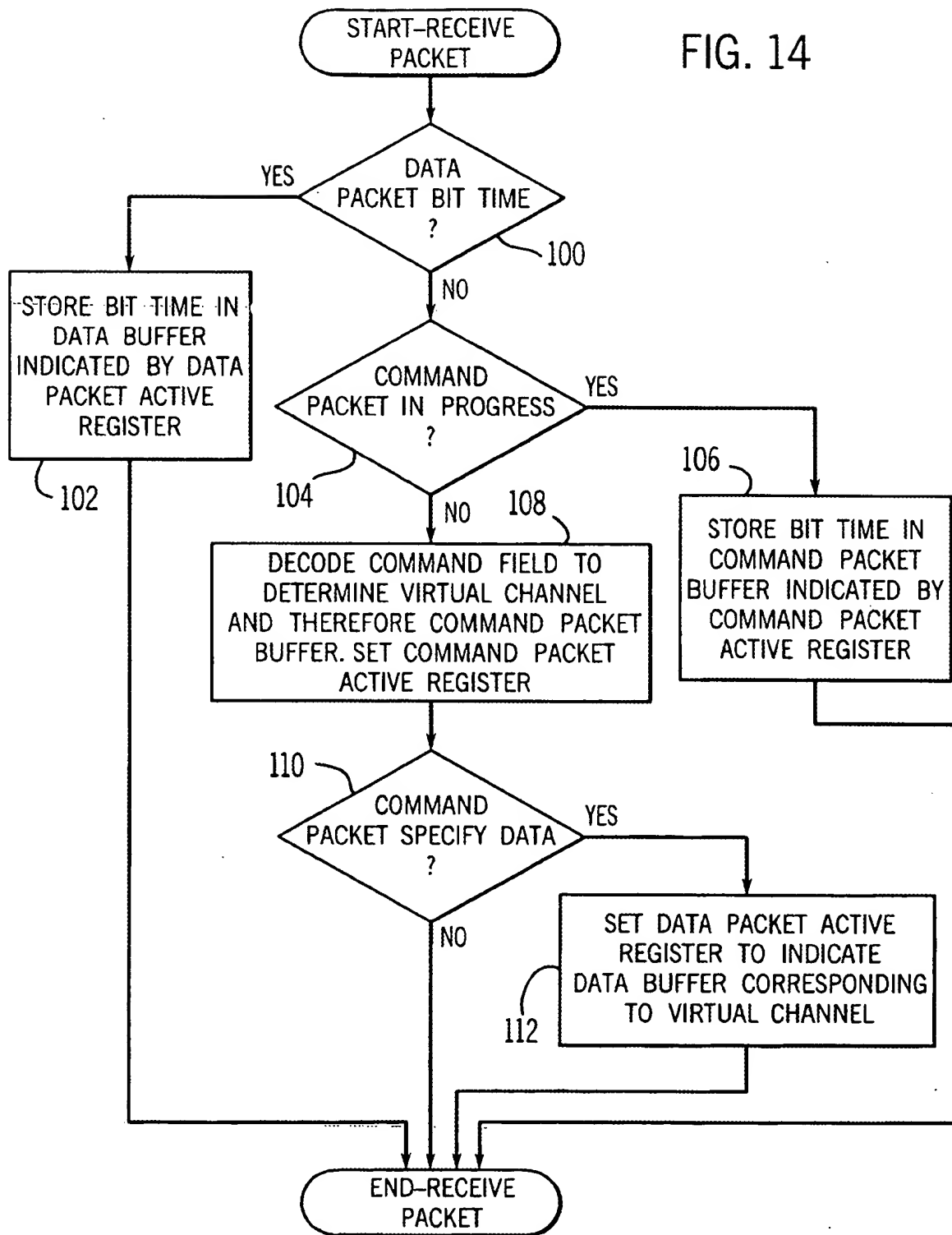


FIG. 15

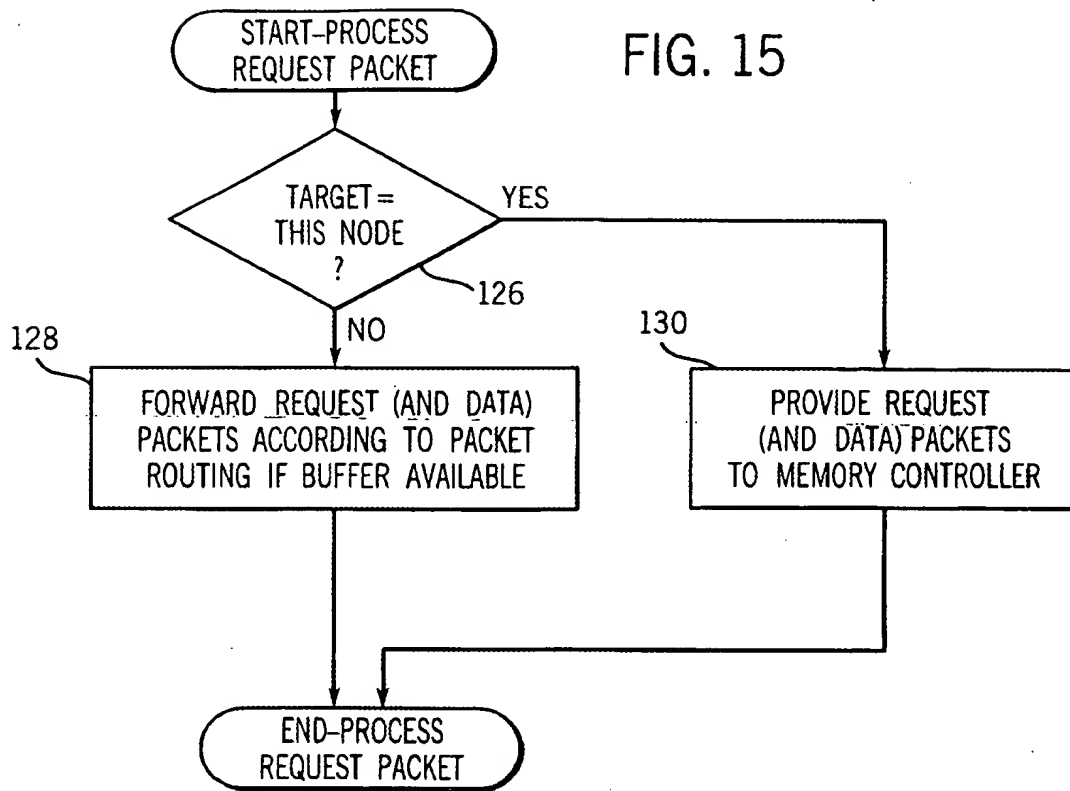
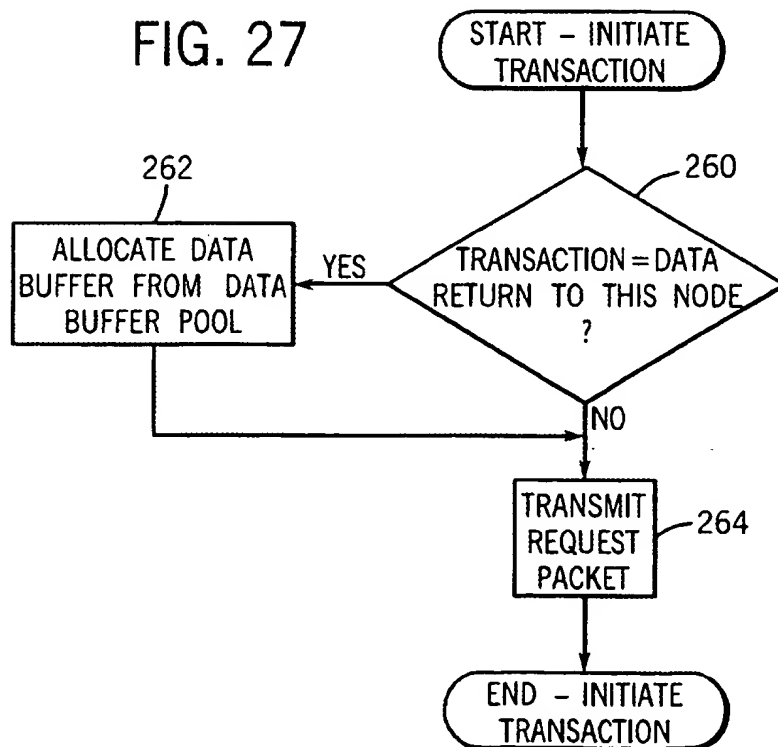


FIG. 27



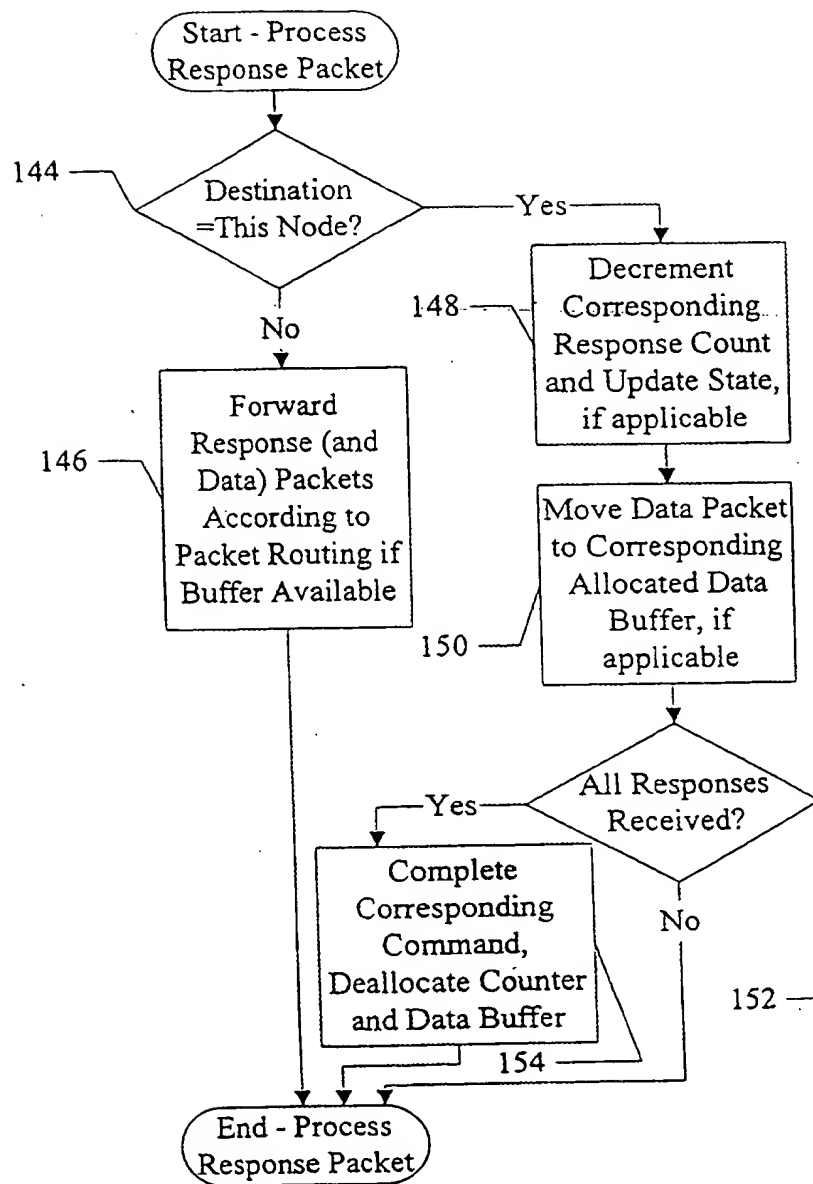


Fig. 16

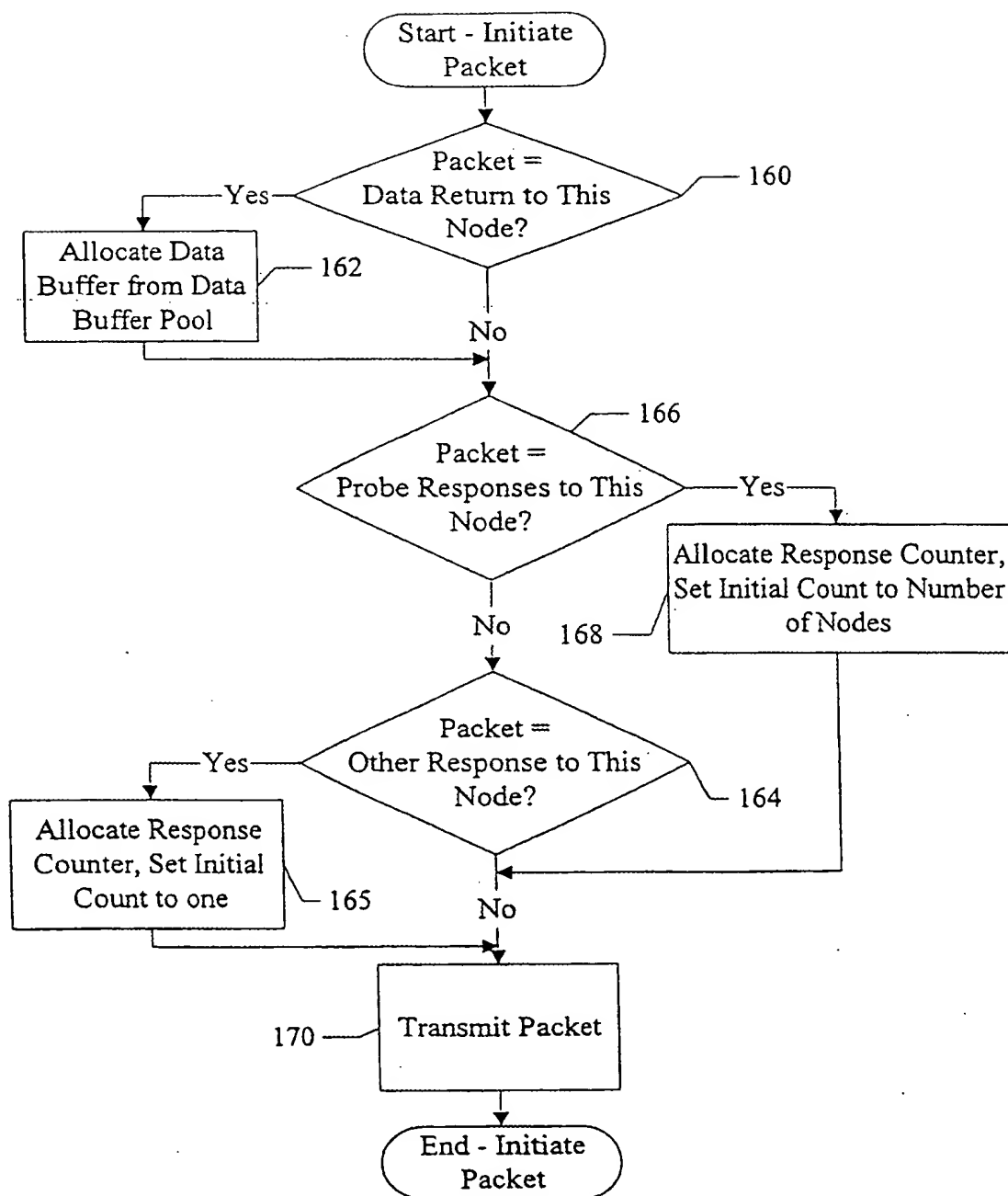
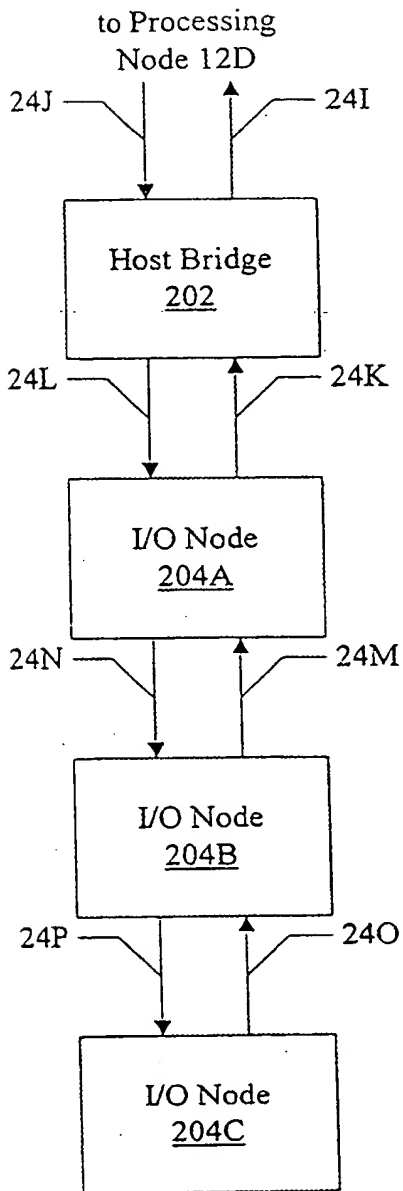


Fig. 17



200

Fig. 19

<u>CMD Code</u>	<u>VChan</u>	<u>Command</u>	<u>Packet Type</u>
000000	–	Nop	Info
000001	–	Reserved	–
000010	NPC	Flush	Request
000011	–	Reserved	–
0001xx	–	Reserved	–
x01xxx	NPC or PC	Wr(Sized)	Request / Address / Data
01xxxx	NPC	Read(Sized)	Request / Address
100xxx	–	Reserved	–
110000	R	RdResponse	Response / Data
110001	–	Reserved	–
110010	–	Reserved	–
110011	R	TgtDone	Response
11010x	–	Reserved	–
11011x	–	Reserved	–
11100x	–	Reserved	–
11101x	PC or NPC	Broadcast	Request / Address
111100	PC	Fence	Request
111101	–	Reserved	–
111110	–	Reserved	–
111111	–	Sync	Info

210 ↗

FIG. 20

Bit Time	7	6	5	4	3	2	1	0
0	SeqID[3:2]		CMD[5:0]					
1	Pass PW	SeqID[1:0]		UnitID[4:0]				
2				SrcTag[4:0]				
3	Addr[7:2]							
4	Addr[15:8]							
5	Addr[23:16]							
6	Addr[31:24]							
7	Addr[39:32]							

FIG. 21

Bit Time	7	6	5	4	3	2	1	0
0			CMD[5:0]					
1	Pass PW			UnitID[4:0]				
2				SrcTag[4:0]				
3								

FIG. 22

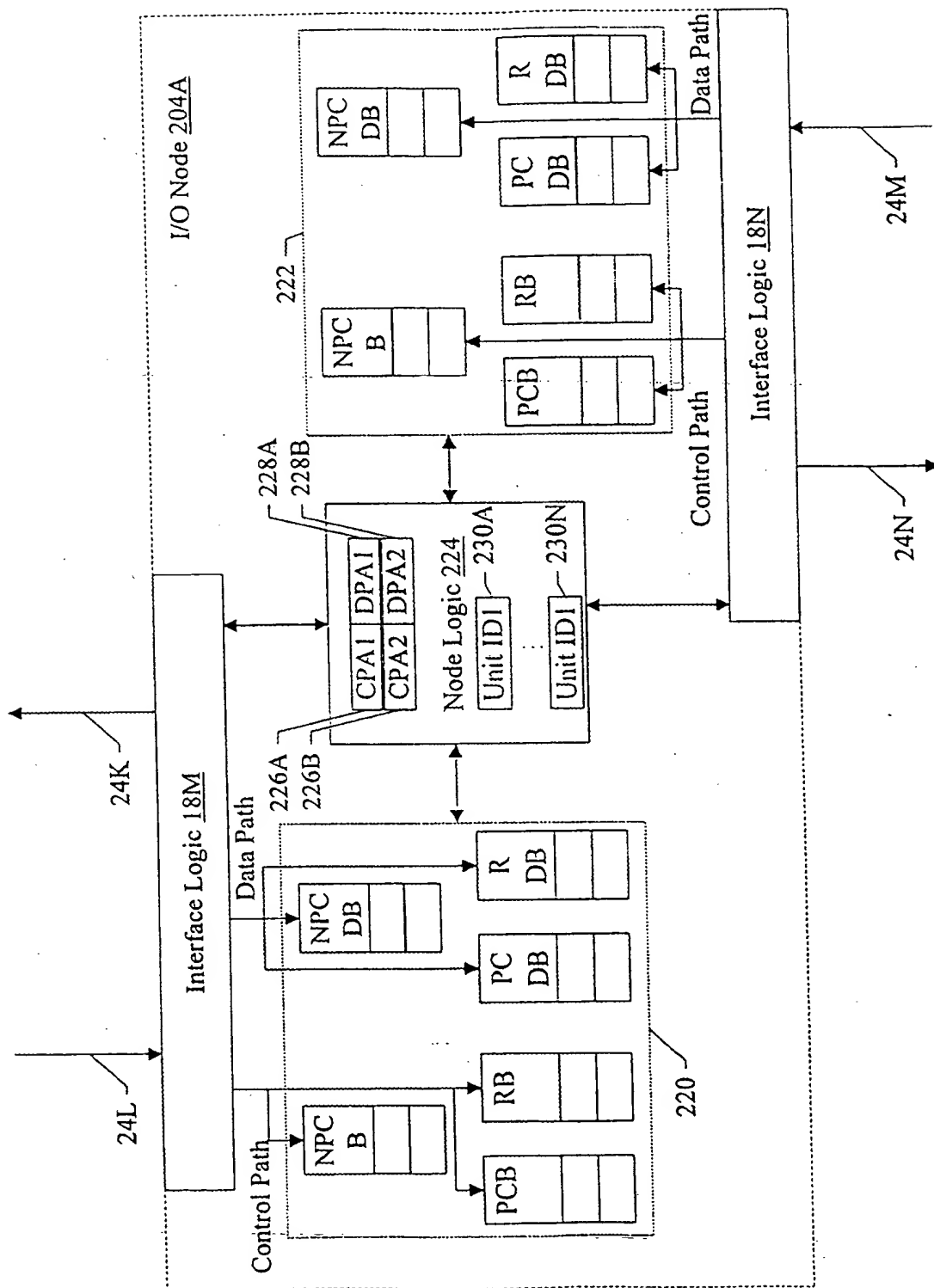


Fig. 23

FIG. 24

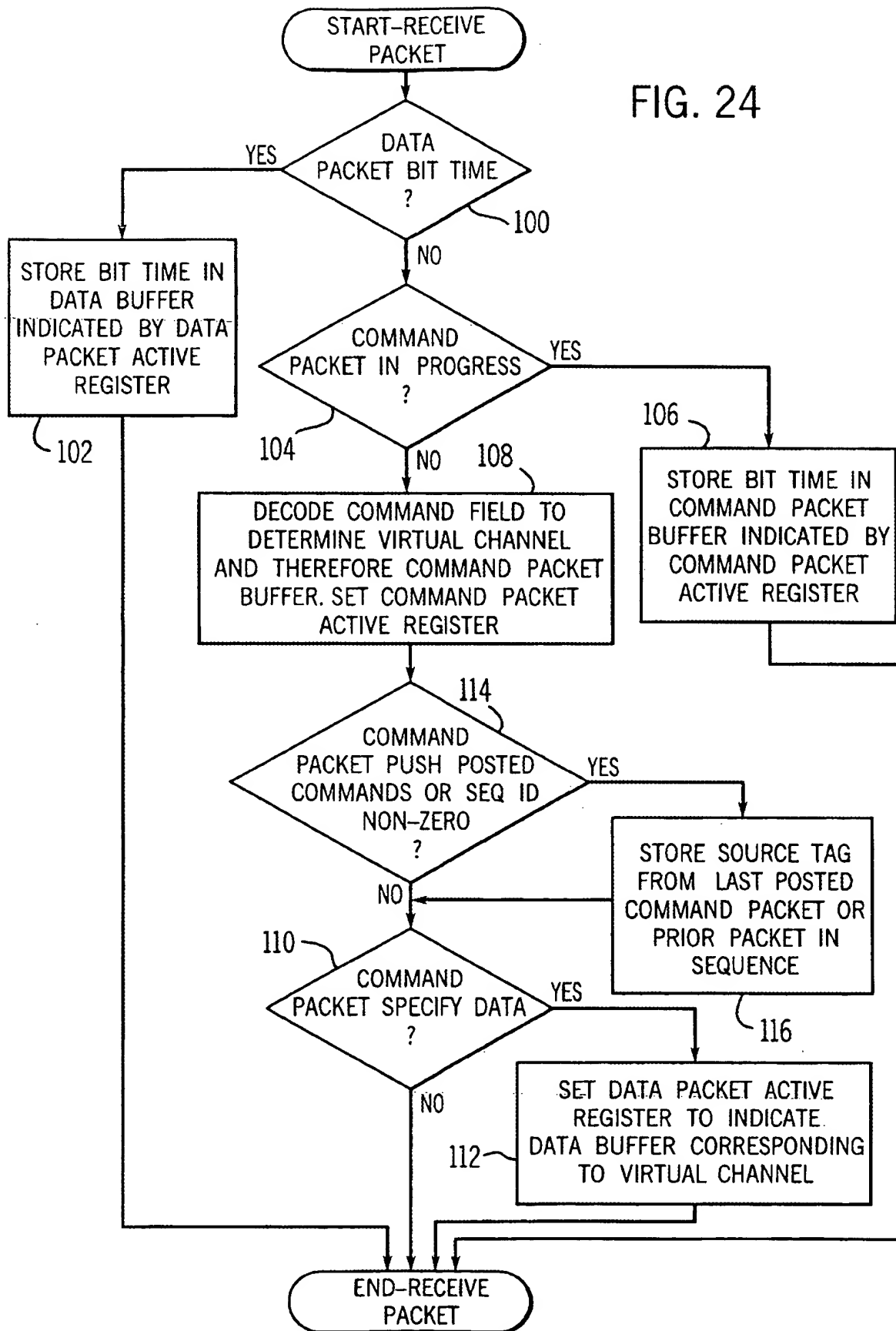


FIG. 25

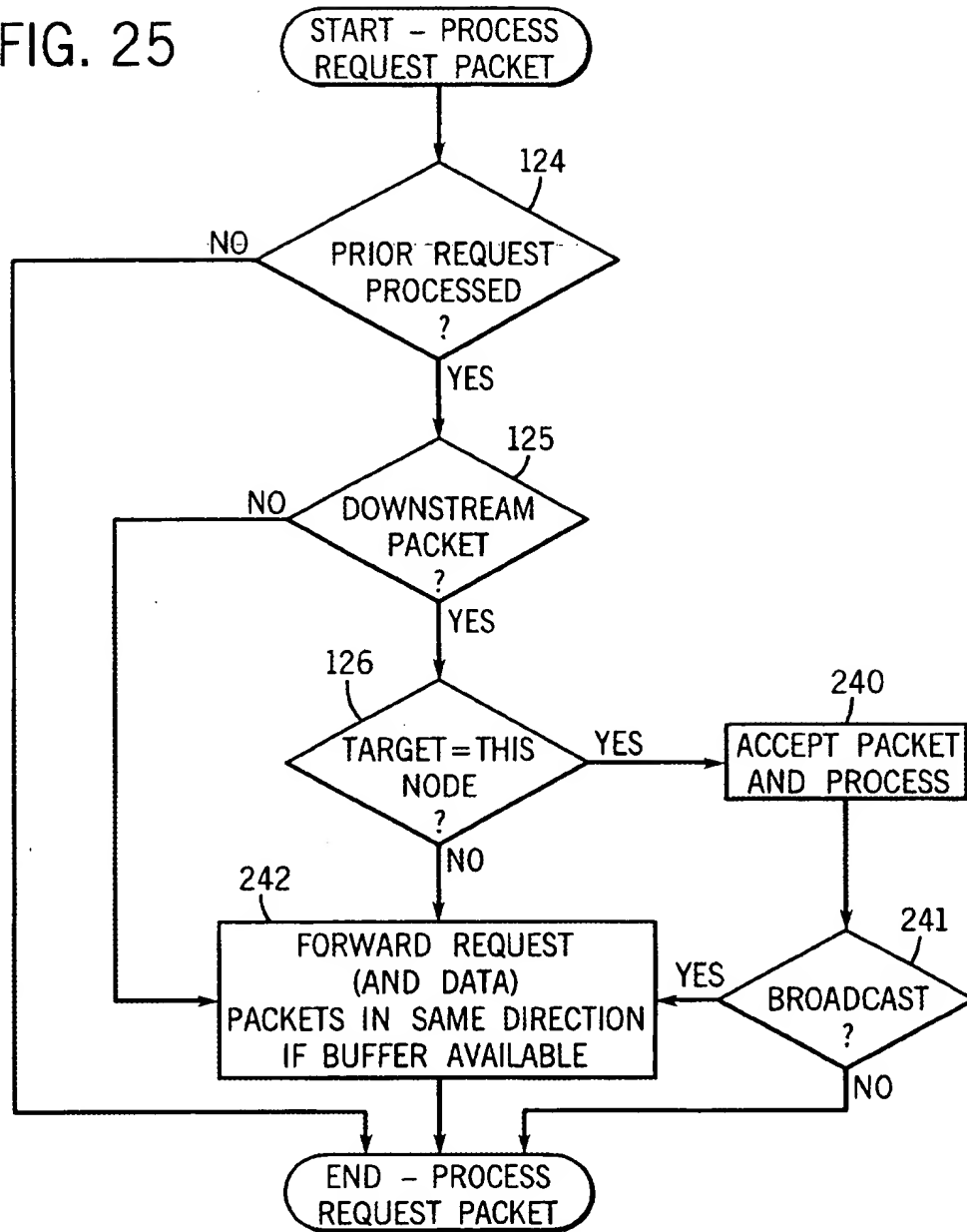


FIG. 26

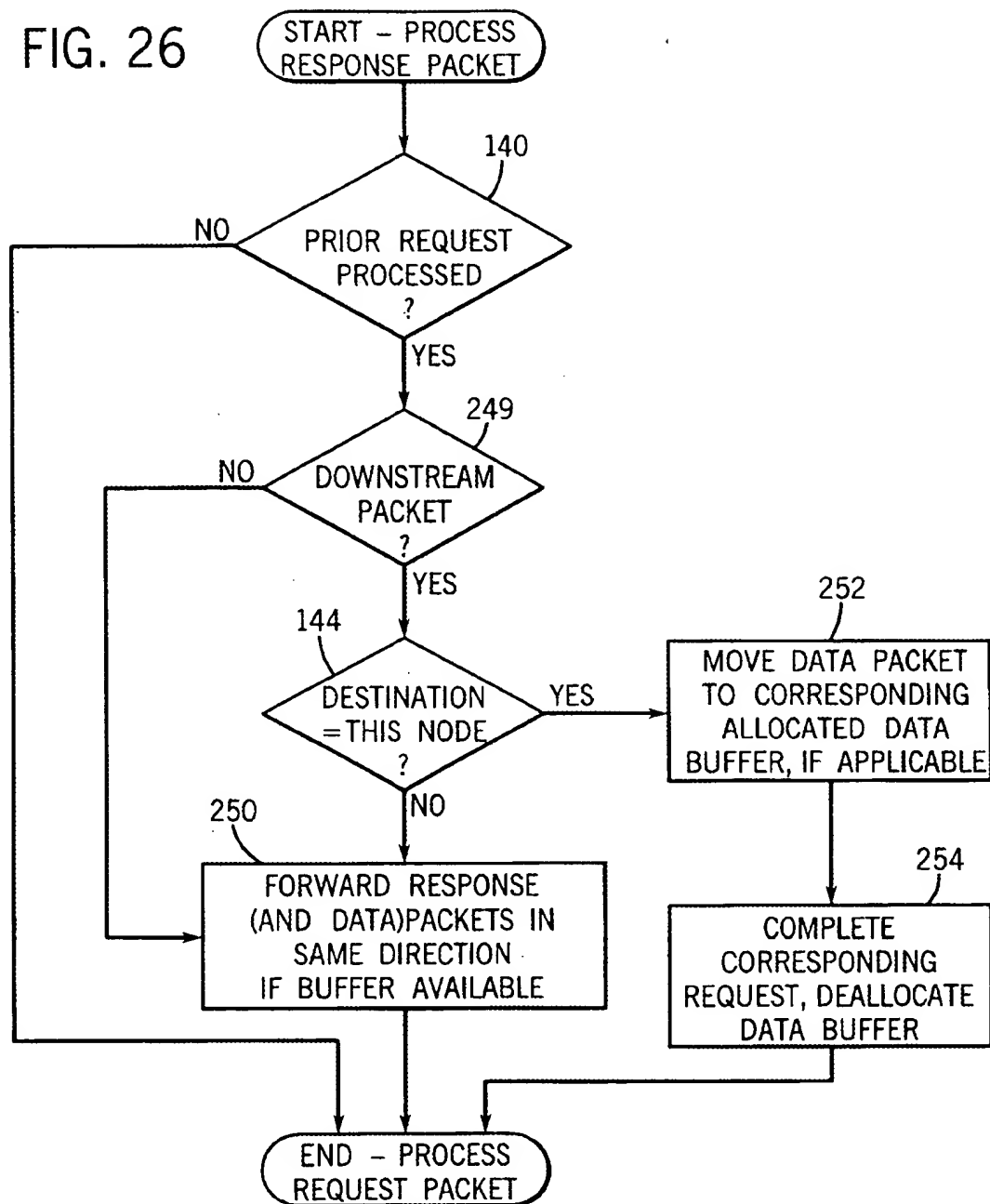


FIG. 28

Request 1 TYPE	Request 2 TYPE	WAIT REQUIREMENTS
272 MEMORY WRITE	MEMORY WRITE	1. Req2 MUST WAIT FOR TgtStart1. 2. SrcDone2 MUST WAIT FOR TgtDone1. 3. TgtDone2 ON THE NON-COHERENT LINK (IF REQUIRED) MUST WAIT FOR TgtDone1.
274 MEMORY WRITE	MEMORY READ	1. Req2 MUST WAIT FOR TgtStart1. 2. TgtDone2 ON THE NON-COHERENT LINK MUST WAIT FOR TgtDone1.
MEMORY READ	MEMORY REQUEST	Req2 MUST WAIT FOR TgtStart1.
MEMORY WRITE	I/O REQUEST OR INTERRUPT	Req2 MUST WAIT FOR TgtStart1.
MEMORY READ	I/O REQUEST	Req2 MUST WAIT FOR TgtStart1.
MEMORY WRITE	FLUSH	TgtDone2 ON THE NON-COHERENT LINK MUST WAIT FOR TgtDone1. (FLUSH DOES NOT CAUSE ANY REQUESTS TO BE ISSUED TO THE COHERENT FABRIC.)
MEMORY READ	FLUSH OR INTERRUPT	NO WAIT REQUIREMENTS
MEMORY WRITE	RESPONSE	Response2 MUST WAIT FOR TgtDone1.
MEMORY READ	RESPONSE	Response2 MUST WAIT FOR TgtStart1.
I/O REQUEST	MEMORY REQUEST	Req2 MUST WAIT FOR TgtStart1.
I/O REQUEST	I/O REQUEST OR INTERRUPT	Req2 MUST WAIT FOR TgtStart1.
I/O REQUEST	FLUSH	TgtDone2 ON THE NON-COHERENT LINK MUST WAIT FOR TgtStart1. (FLUSH DOES NOT CAUSE ANY REQUESTS TO BE ISSUED TO THE COHERENT FABRIC.)
I/O REQUEST	RESPONSE	Response2 MUST WAIT FOR TgtStart1.
FLUSH	ANYTHING	NO WAIT REQUIREMENTS
RESPONSE	ANYTHING	NO WAIT REQUIREMENTS
FIXED /NON VECTORED INTERRUPT	RESPONSE	Response2 MUST WAIT FOR ALL BROADCAST MESSAGE RESPONSES TO BE RECEIVED.
FIXED /NON VECTORED INTERRUPT	ANYTHING BUT RESPONSE	NO WAIT REQUIREMENTS
LPA INTERRUPT	ANYTHING	NO WAIT REQUIREMENTS
SysMgt	ANYTHING	NO WAIT REQUIREMENTS
FENCE	POSTED REQUEST	Req2 MUST WAIT FOR FENCE TO BE RETIRED.
FENCE	ANYTHING NONPOSTED	NO WAIT REQUIREMENTS
POSTED MEMORY WRITE	FENCE	Req2 MAY BE RETIRED WHEN TgtDone1.
270 POSTED I/O WRITE	FENCE	Req2 MAY BE RETIRED WHEN TgtStart1.
ANYTHING NONPOSTED	FENCE	NO WAIT REQUIREMENTS